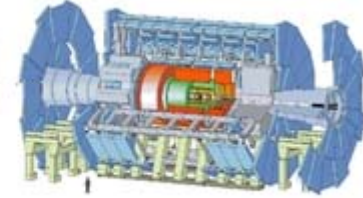




3D Workshop Nov 2008



the **ATLAS Experiment**



Life Cycle of TAG data

Florbela Viegas



TAGS data features



- TAGS data is a collection of « POOL Collections »
- Collections are independent from each other, basically two tables of « DATA » and « LINKS »
- The data access is:
 - Insert only activity by an upload service
 - Index the data for reading
 - Read-only access from several services
- The last upload of collections created one collection per parsing of dataset name like:
 - T0 task name:
1813367_fdr08_run2.0052280.physics_Jet.merge.TAG.o3_f47_m26_bnl.upload.task._0023.job.2.log (Run 52280)
 - Collection: **fdr08_run2_physics_Jet_o3_f47_m26**
- These logical names are mapped in a dictionary table, to the corresponding DATA and LINKS Oracle tables.



TAGS data features - II



- TAGS data is expected to grow at 200Hz rate, when events are being produced, which amount to 11.2 TB per nominal year.
- This accounts to holding two reprocessing passes and a yearly reconstruction of all data.
- Simulations have been made for sites requesting only one year of data
- All calculations are here:

<https://twiki.cern.ch/twiki/bin/view/Atlas/DatabaseVolumes>



A TAG's life



- Decisions have not been made yet regarding the deletion of data
- At CERN the assumption is that all data is kept BUT still there is the issue of what to do if there are more than 2 reprocessing passes.
- Reproducibility of queries is a big issue, although it hasn't been defined thoroughly what is meant by this.
- Assumption that a physicist will want only latest version of data is reasonable, but on-demand access to old data should be kept possible.



A TAG's life - II



- Measures are being taken to make sure the management of data is transparent and facilitated:
 - PL/SQL procedure is called when collection is created: used now for partitioning the table by run number, and indexing the table a priori
 - Checking that a collections exists does not access the data dictionary, it relies instead on the POOL_COLLECTIONS table: gives us freedom to redefine the collections as the objects we wish (views, synonyms, etc.)
 - PL/SQL procedure is « hooked » into the T0 upload system on the opening and closing of a run: (can be used for a posteriori indexing, making data read-only, available to users, etc.)
 - Structure of upload service can be changed to make the collections with different granularity, and more information for management and identification of run passes, for example.



Summary



- Many scenarios for data management, but only data volumes are confirmed
- Care is being taken for flexibility of management and « hooks » in the code, many thanks to Marcin Nowak and Luc Goosens for this ongoing work.
- Many of these operational issues will be tested and refined in the next TAG upload exercise, expected at end of November 2008.